Relations

Adapted from a handout written by Dr. Bob Plummer

Relations are a fundamental concept in discrete mathematics, used to define how sets of objects relate to other sets of objects. Not only do they provide a formal way of being able to talk about such relationships, they also provide the most widespread model used in modern commercial database systems. Understanding relations from a mathematical perspective not only gives you an important modeling tool, but also gives you the foundational theory used in a number of applications including relational database management systems, task scheduling systems, and methods to solve various optimization problems.

To explore what relations are, let's begin by considering the following set *G* Greek deities:

G = {Zeus, Apollo, Cronus, Poseidon}

As you may know, Zeus is the father of Apollo, Cronus is the father of Poseidon, and Cronus is also the father of Zeus. So some of the elements of *G* that satisfy the "is the father of" relation with respect to others Notice that in this case, the elements that are related are both from the same set, G, and that the relationship is "one way:" if X is the father of Y, then Y is not the father of X. If we had another set, H, of female deities, then some members of G might bear the "is married to" relation to members of H, and that relationship would also be true in the other direction. We will formalize all of these ideas, starting with some definitions.

Definitions

A **sequence** of objects is a list of these objects in some order. Sequences may be finite or infinite.

A finite sequence is called a **tuple**. A sequence with **k** objects is called a **k-tuple**.

An **ordered pair** is a 2-tuple; that is, an ordered sequence of two elements. We write ordered pairs in parentheses, for example **(a, b)**, and we call **a** the first element and **b** the second element of the pair.

The **Cartesian product** or **cross product** of two sets A and B, written **A × B**, is the set of all ordered pairs wherein the first element is a member of A and the second element is a member of B.

A **binary relation** *R* between two sets *A* and *B* (which may be the same) is a subset of the Cartesian product $A \times B$. If element a ∈ A is related by *R* to element b ∈ B, we denote this fact by writing **(a, b) ∈ R**, or alternately, by **a R b**. We say that **R is a relation on A and B**.

A **relation on a set A** is a subset of A × A.

A good way to think of a binary relation is that it is a way to designate that of all the ordered pairs in the cross product of two sets, some are "interesting" because there is a certain relationship between them. We often name relations with capital letters, but some relations, such as "less-than" have their own symbols, like <.

What we defined above is a **binary relation** because it operates on ordered pairs. We can also define **unary relations,** which operate on single elements, or **ternary relations**, which operate on ordered triples. In general an ***n*-ary relation** will operate on *n*-tuples. Formally, we can express this as:

Definition

An ***n*-ary relation** on the sets $A_1, A_2, \ldots, A_n$ is a subset of $A_1 \times A_2 \times \ldots \times A_n$. The sets $A_1, A_2, \ldots, A_n$ are called the **domains** of the relation, and n is called its **degree**.

Example 1

Let's consider the "is the father of" relation (which we will denote by F) on the set G □G. Here is the cross product:

G × G = { (Zeus, Zeus), (Zeus, Apollo), (Zeus, Cronus), (Zeus, Poseidon),
(Apollo, Zeus), (Apollo, Apollo), (Apollo, Cronus), (Apollo, Poseidon),
(Cronus, Zeus), (Cronus, Apollo), (Cronus, Cronus), (Cronus, Poseidon)
(Poseidon, Zeus), (Poseidon, Apollo), (Poseidon, Cronus), (Poseidon, Poseidon) }

But of the set G × G, only a subset satisfies the "is the father of" relation. Thus, applying the F relation to G × G yields the set:

F = {(Zeus, Apollo), (Cronus, Poseidon), (Cronus, Zeus)}

which we could also write as Zeus F Apollo, Cronus F Poseidon, and Cronus F Zeus.

Graphs and Relations

Graphs are a general representation for expressing many-to-many relationships. The easiest way to understand the definition of a graph is to look at a picture. Here are three examples of graphs:

Intuitively, we get the idea that a graph is a bunch of points connected by lines. The formal definition conveys this concept more precisely:

Definitions

A **graph** is an ordered pair (*V*, *E)* where

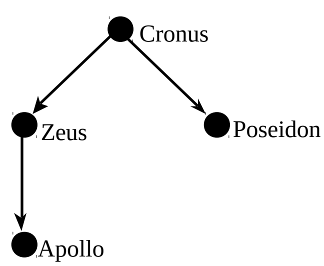
(i) *V* is a non-empty set of **nodes** or **vertices** (dots)

(ii) *E* is a set of **arcs** or **edges**, where each edge is a pair of vertices.

An **undirected graph** is a graph where the edges are unordered pairs.

A **directed graph** is a graph where the edges are ordered pairs.

Graphs are very useful structures that we will be re-examining later in the class as well. The first use we will put graphs to is to represent the family relation described by the "father of" relation.



Notice that this graph has arrows rather than lines connecting the nodes, indicating that this is a directed graph. Directed graphs are very useful for representing binary relations, where the relation *a R b* is represented by drawing an arrow from *a* to *b*.

Properties of Relations

Definitions

A relation R is called **reflexive** on a set S if for all $x \in S$, $(x, x) \in R$.

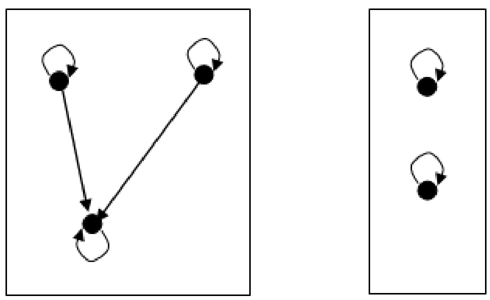
A relation R is called **irreflexive** on a set S if for all $x \in S$, $(x, x) \notin R$.

A relation R is **symmetric** on a set S if for all $x \in S$ and for all $y \in S$, if $(x, y) \in R$ then $(y, x) \in R$.

A relation R is **antisymmetric** on a set S if for all $x \in S$ and for all $y \in S$, if $(x, y) \in R$ and $(y, x) \in R$, then x = y

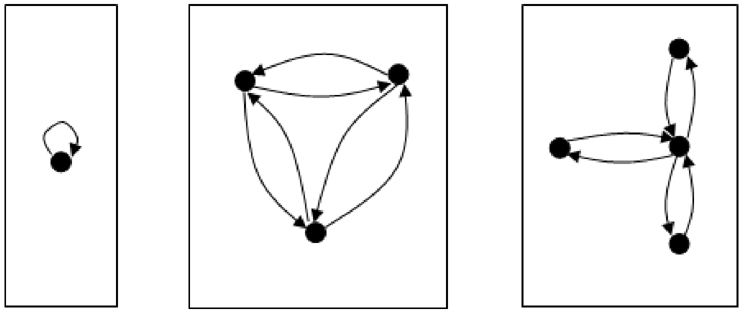
A relation R is **transitive** on a set S if for all $x, y, z \in S$, if $(x, y) \in R$ and $(y, z) \in R$, then $(x, z) \in R$.

In a graph of a **reflexive** relation, every node will have an arc back to itself. For example, the relations below, represented as graphs, are reflexive:

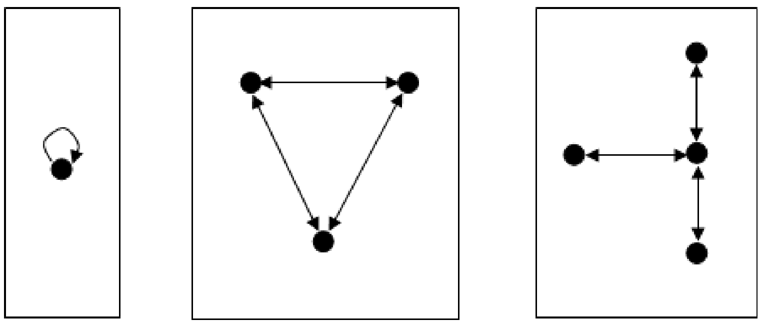


Note that **irreflexive** says more than just **not reflexive**. Neither of the relations pictured above would be reflexive if we removed just one of the loops, such as the loop from *b* back to itself. To make them **irreflexive**, we would have to remove all such loops. The "less than" relation < is irreflexive on the integers, since no integer is less than itself.

The "is a sibling of" relation is **symmetric**. If Zeus is a sibling of Poseidon, then Poseidon is a sibling of Zeus. It is easy to tell if a relation is symmetric by looking at its graph. The relation is symmetric if every node x in a graph that is connected by an arc to another node y, has an arc from node y back to node x. The relations below are symmetric.



Sometimes, as a notational shorthand, we will combine the two arcs connecting two nodes in both directions in a graph into a single line in the graph that has an arrowhead at both ends. Thus the graphs of the relations above would be drawn as:



A relation is **not symmetric** if there is at least one pair (x,y) in the relation such that (y, x) is not in the relation. That is, for a relation to be symmetric, it has to be true for *all* x and y that x R y implies y R x, not just a handful. Similarly, antisymmetry is **not** the same as being not symmetric. You'll explore this on the first problem set.

The “is the father of” relation is antisymmetric. If Zeus is the father of Apollo, then certainly Apollo is not the father of Zeus. In terms of a directed graph, a relation is antisymmetric if whenever there is an arrow going from an element to another element, there is not an arrow from the second element back to the first.

Transitivity is a familiar notion from both mathematics and logic. The “less-than” relation (<) is transitive. If $x < y$, and $y < z$, then it must be true that $x < z$.

Equivalence Relations

The properties of relations are sometimes grouped together and given special names. A particularly useful example is the equivalence relation.

Definitions
A relation that is reflexive, symmetric, and transitive on a set *S* is called an **equivalence relation** on *S*.

If R is an equivalence relation on S and $x \in S$, then the equivalence class of x with respect to R is denoted $[x]_R$ and defined by $[x]_R = \{y \mid y \in S \text{ and } x\ R\ y\}$. Since R is reflexive, $x \in [x]$.

It will be seen that if R is an equivalence relation on S and x R y, then $[x]_R = [y]_R$; that is, these are two notations for the same set of elements of S. When we write $[x]_R$, x is called a representative of the equivalence class. If it is clear what relation is being discussed, we often drop the subscript R and just write [x].

Example 2

Suppose *R* is the relation on a set of strings *S* such that a *R* b holds if and only if length(a) = length(b), where length(x) means the length of string x.

Since length(a) = length(a), we can say a *R* a and therefore, *R* is reflexive. Suppose a *R* b, so length(a) = length(b). Then, length(b) = length(a) so *R* is also symmetric. Finally, suppose a *R* b and b *R* c; this means length(a) = length(b) and length(b) = length(c). We see that length(a) = length(c) so *R* is also transitive and is thus an equivalence relation.

Graphs of equivalence relations are divided into "islands" of interconnected nodes, with each separate sub-graph containing a group of nodes that are mutually equivalent. In the string example, each island would contain strings of equal length. These islands are the **equivalence classes.** Here are some definitions that help sharpen this notion.

Definitions

Two sets A and B are **disjoint** if they have no common elements; that is, if $A \cap B = \emptyset$.

A family of sets $A_1, A_2, \ldots, A_n$ is **pairwise disjoint** if for all $i, j \leq n$, if $i \neq j$, then A_i and A_j are disjoint.

A **partition** of a non-empty set *S* is a collection of pairwise disjoint non-empty subsets of *S* that have *S* as their union.

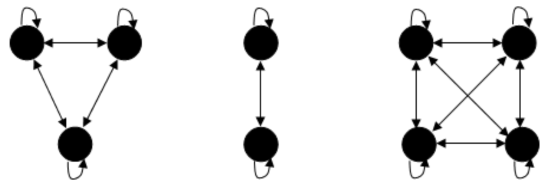
The last definition can also be written like this:

Let S be a non-empty set. A **partition** of S is a collection D of non-empty subsets of S such that

(i) if $P, Q \in D$ and $P \neq Q$, then $P \cap Q = \emptyset$; and
(ii) $\bigcup_{P \in D} P = S$.

The notation in condition (ii) means that the union of all the sets in the partition D is S. Given these definitions, we see that if R is an equivalence relation on S, the equivalence classes of R form a partition of S.

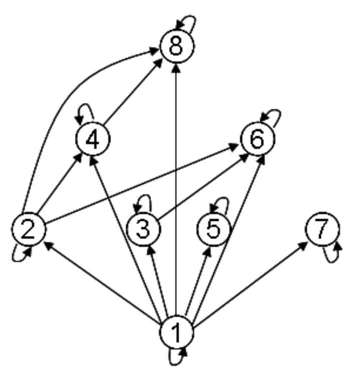
Examples of equivalence relations on a set of people include "same age as" and "born in the same month as." If we have nine people, with three born in March, two born in May, and four born in December, the graph of the relation "born in the same month as" would appear as follows:



Partial Orderings

Definitions
A relation *R* that is reflexive, <u>anti</u>symmetric, and transitive on a set *S* is called a **partial ordering** on *S*.
A set *S* together with a partial ordering *R* is called a **partially ordered set** or **poset**.

As a small example, let S = {1, 2, 3, 4, 5, 6, 7, 8}, and let R be the binary relation "divides." So (2,4) ∈ R, (2, 6) ∈ R, etc. Using | as the symbol for "divides", we see that R is reflexive, since x | x; R is transitive since x | y and y | z implies x | z; and R is antisymmetric since we never have x | y and y | x where x ≠ y. Thus S and R form a poset. The following graph represents R:



The loops on each vertex show reflexivity; the arrows between nodes like 2, 4, and 8 show transitivity; and that fact that no two nodes have arrows to and from each other shows antisymmetry.

Another thing to notice about this relation is that only certain pairs of integers in S × S are related. We know that 4 | 8, so (4, 8) is in the relation, and 4 comes "before" 8 in the ordering. Likewise 3 comes before 6, and 1 comes before all the others. This relation does not give an ordering of nodes like 3 and 7, however, since $3 \nmid 7$ and $7 \nmid 3$. That is why we say that the ordering is **partial**. The relation gives an ordering of some, but not all, pairs of vertices.

What if we wanted to list all the vertices in an order that does not violate the ordering provided by R? We know that 1, 2, 3, 4, 5, 6, 7, 8 would work. But so would 1, 5, 3, 2, 7, 4, 6, 8, since whenever $(x, y) \in R$, x comes before y in the list. We will return to this idea below.

Example 3

Problem: Show that $\geq$ is a partial ordering on the set of integers.

Solution: Since $a \geq a$, this relation is reflexive. If $a \geq b$ and $b \geq a$, then $a = b$ which shows this relation is antisymmetric. If $a \geq b$ and $b \geq c$, then $a \geq c$ so this relation is transitive. Thus, $\geq$ is a partial ordering on the set of integers.

We normally write $(N, \geq)$ to mean the poset that we get from the integers and the relation $\geq$. That is a useful notation, as long as you remember that the second member of the poset is a relation, that is, a set of ordered pairs (a, b) that satisfy $a \geq b$.

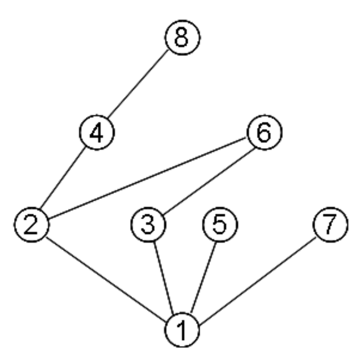
Hasse Diagrams

The graphs of partial orderings can be fairly complex. (Being both reflexive and transitive produces a lot of arcs.) One way to simplify these graphs is to use a special kind of representation known as a Hasse diagram.

The general procedure for creating a Hasse diagram is as follows:

1. Draw the directed graph for the relation
2. Remove all the loops at each node, which must be there for reflexivity
3. Remove all edges that are there for transitivity; that is, wherever there are three nodes x, y, and z with edges from x to y and from y to z, remove the edge between x and z.
4. Arrange each edge so that its initial node is below its terminal node as indicated by the directed edges
5. Remove all arrows on the directed edges (since all edges point upward toward their terminal node)

Here is a Hasse diagram for the poset discussed above:



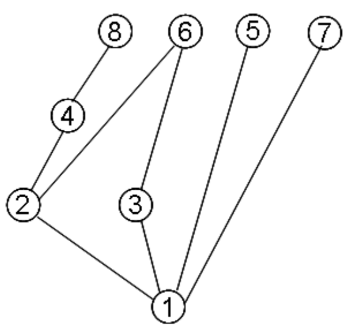
Because posets are at least partially ordered, it is possible to pick out some extremal elements. We define them as follows.

Definitions

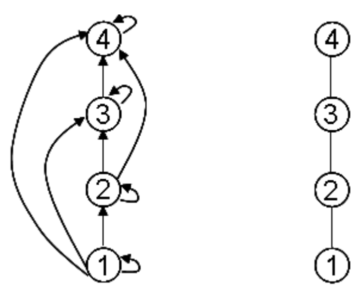
An element a of a poset (*S*, *R*) is **maximal** if it is not less than any element of the poset; that is, a is maximal if there is no b ∈ S such that a R b.

An element a of a poset (*S*, *R*) is **minimal** if it is not greater than any element of the poset; that is, a is minimal if there is no element b ∈ S such that b R a.

In the example above, 5, 6, 7, and 8 are maximal elements, and 1 is the only minimal element**.** Extremal elements are easy to spot in Hasse diagrams, and in this case they would have been even easier to spot if we had drawn the following, equivalent, diagram:



Now we consider the partial ordering {(a, b) | a ≤ b} on { 1, 2, 3, 4 }. The directed graph and a Hasse diagram are shown below.



Definitions

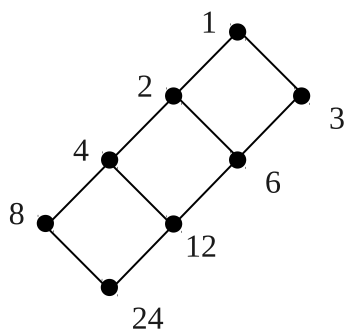
Suppose *R* is a partial ordering on a set *A*. Elements a and b of *A* are said to be **comparable** if and only if, either a *R* b or b *R* a. Otherwise a and b are **noncomparable**.

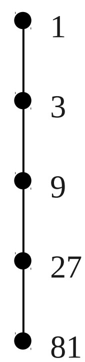
A **total ordering** is a special case of a partial ordering where every pair of elements is comparable. Since a total ordering is a partial ordering, it is reflexive, antisymmetric, and transitive.

The Hasse diagram for a total ordering is about as simple as possible: it's a straight line. The example given above for ≤ is a total ordering.

Example 4

The Hasse diagram for the partial ordering {(a,b) | a mod b = 0 } on the set of {positive divisors of 24} is given below. (a mod b is the remainder when a divided by b. So a mod b = 0 means b divides a evenly.) This is not a total ordering because 6 and 4 (for example) are not comparable. The second diagram for the partial ordering {(a, b) | a mod b = 0 } on the set of {positive divisors of 81} is a total ordering. (You are correct if you are thinking that we defined this relation in terms of mod rather than using "divides" just to make 1 come out at the top this time.)





We present two further definition concerning orderings:

Definitions

A relation *R* that is irreflexive, antisymmetric, and transitive on a set *S* is called a **strict ordering** on *S*.

If R is a strict ordering of S such that exactly one of a R b, b R a, or a = b holds, R is called a **strict total ordering** on A.

Bibliography

Many good math books will provide coverage of relations. Some excellent references are:

J. A. Anderson, *Discrete Mathematics with Combinatorics, 2nd Ed.*, New Jersey: Pearson Prentice Hall, 2004.

Ethan D. Bloch, *Proofs and Fundamentals. A First Course in Abstract Mathematics, 2nd Ed*. New York: Springer, 2011.

K. Rosen, Discrete Mathematics and its Applications, 6th Ed., New York: McGraw-Hill, 2007.

With thanks to those involved with previous offerings of CS103 and its predecessors.